Retinal algorithm for heavy-ion tracking in single event effects localization*

Wendi Deng,[1] Jinchuan Wang,[1] Huipeng Pan,[1] Wei Zhang,[1] Jiansong Wang,[2] Fuqiang Wang,[2,3] Zili Li,[2,†] and Renzhuo Wan[1,‡]

[1]*School of Electronics and Electrical Engineering, Wuhan Textile University, Wuhan 430000, China*
[2]*School of Science, Huzhou University, Huzhou 313000, China*
[3]*Strong-coupling Physics International Research Laboratory, Huzhou University, Huzhou 313000, China*

This study presents a real-time tracking algorithm derived from the retinal algorithm, designed for the rapid, real-time tracking of straight-line particle trajectories. These trajectories are detected by pixel detectors to localize single event effects in two-dimensional space. Initially, we developed a retinal algorithm to track the trajectory of a single heavy ion and achieved a positional accuracy of 40 μm. This was accomplished by analyzing trajectory samples from the simulations using a pixel sensor with a 72 × 72 pixel array and an 83 μm pixel pitch. Subsequently, we refined this approach to create an iterative retinal algorithm for tracking multiple heavy-ion trajectories in single events. This iterative version demonstrated a tracking efficiency of over 97%, with a positional resolution comparable to that of single-track events. Furthermore, it exhibits significant parallelism, requires fewer resources, and is ideally suited for implementation in field-programmable gate arrays on board-level systems, facilitating real-time online trajectory tracking.

Keywords: Single-event effects, Retinal algorithm, Iterative retinal algorithm, Heavy ion, Particle tracking

I. INTRODUCTION

In recent years, there have been significant concerns over the performance degradation of electronic devices caused by radiation, along with various reinforcement measures aimed at avoiding these adverse effects. As microelectronics continue to advance towards greater complexity and miniaturization, the failure mechanisms of space electronic systems induced by natural radiation environments have also become more complex and severe [1–4]. Currently, the main effects of radiation-induced performance degradation in electronic systems that have been identified are: total ionizing dose effects (TID) [5], displacement damage effects [6, 7], and single-event effects (SEE) [8]. Space missions have indicated that SEEs are an important cause of anomalies and failures in spacecraft operations [9]. Satellites from various countries have suffered from the harm caused by SEE, which originates from ionizing cosmic rays with high energy [10], leading to two types of errors in spacecraft electronics. Soft errors are mainly caused by a single-event upset (SEU) [11], single-event transient (SET) [12], and single-event function interrupt (SEFI), which can change the logical state of a storage unit and lead to device dysfunction. However, the disorder is always temporary and can be restored. Hard errors include single-event latch-up (SEL) [13], single-event burnout (SEB) [14], and single-event gate rupture (SEGR), etc., which can lead to permanent component damage and system failure. Consequently, various methods have been adopted to conduct radiation tests and enhance the radiation resistance of ICs used in spacecraft [15–18].

A common approach for SEE testing is to use heavy-ion beams generated by accelerators on the ground to irradiate electronic devices. This irradiation approach is always employed to assess the overall radiation resistance of devices rather than to evaluate the response of specific areas or modules to heavy-ion impacts. To reinforce a device more effectively against radiation, it is necessary to identify the affected parts that trigger SEE and evaluate their sensitivity to SEE. For the real-time localization of SEE, the capability to promptly and effectively pinpoint the impact of radiation particles on electronic devices under radiation exposure is an important research direction. Therefore, it is imperative to deploy a swift and proficient particle-trajectory tracking algorithm.

Research on tracking algorithms in high-energy physics began quite early and continues to this day [19–22]. In terms of traditional algorithms, Günther et al. [23] enhanced the event processing capacity of the ‘‘forward tracking” algorithm by incorporating single instruction multiple data (SIMD), including the Hough transform and cluster search into crucial components, and ultimately improved the algorithm's event throughput by 60% for CERN's Large Hadron Collider beauty (LHCb) experiment. Hennequin et al. [24] and others proposed a parallel reconstruction algorithm for use in the vertex detectors of the LHCb experiment, achieving ultrafast trajectory reconstruction and tracking and meeting the goal of real-time data processing. In addition, machine-learning-based tracking algorithms have been extensively applied in high-energy physics [25–28]. Ai et al. [29] proposed a deep learning approach to reconstruct the location and orientation of multiple heavy-ion tracks in gaseous drift chambers for a Cooler Storage Ring External Target Facility Experiment (CEE) [30, 31]. Hu et al. [32] proposed a fast multitrack location method based on deep learning to fit each track of the incident heavy ions to SEE terminals at the Heavy Ion Research Facility in Lanzhou (HIRFL).

A series of studies on retinal tracking algorithms have already been conducted. The retinal algorithm [33–36] was designed to locate curved trajectories of charged particles in

* This work was supported by the National Natural Science Foundation of China (No. 12205224), the Research Foundation of Education Bureau of Hubei Province China (No. Q20221703), the National Natural Science Foundation of China (No.12035006, U2032140), and the National Key Research and Development Program of China (No. 2020YFE0202000).

† Corresponding author, zili_li@zjhu.edu.cn
‡ Corresponding author,wanrz@wtu.edu.cn

barrel-shaped track detectors under strong magnetic fields. The results indicate that the tracking efficiency of the retina reaches 95%, and a fixed 2.5 μs is required to process a $t\bar{t}$ event. This demonstrates that the algorithm can fulfill the particle-tracking identification requirements of the CMS experiment in the spatial region of the outer track detector.

Among the studies on SEE localization fields, our previous work [37] proposed a method for directly locating SEE by capturing the projected track of heavy ions using a silicon pixel sensor in a gas detector. One of the key issues to be addressed is the requirement for real-time reconstruction algorithms for heavy ions. Considering that the fast particle trajectory reconstruction capabilities of the retinal algorithm and MAPS chips are becoming prevalent in high-energy physics for precise particle track measurements[38–42], the retinal algorithm was applied to SEE localization. The primary goal is to explore the applicability of the retinal algorithm for tracking linear trajectories within the two-dimensional (2D) space of pixel array sensors and assess the performance of the algorithm by applying it to the reconstruction of simulated heavy-ion tracks before its implementation on FPGAs in a realistic environment.

In the simulation, 25 MeV/u Kr ions were chosen to pass through a gas chamber composed of $Ar : CO_2$(70:30) at 0.8 atm and 300 K, resulting in the production of ionized electrons along the trajectory. An electric field of 300 V/cm was applied perpendicular to the pixel sensor plane in the gas chamber to generate electrons that drift onto the pixel anode of a Topmetal sensor [43–45]. Subsequently, the collected electrons form a linear particle trajectory of the incident particles, which serve as input samples for the algorithm. This study introduces two variations of the retinal algorithm designed for tracking both single and multiple trajectory samples, referred to as the original retinal algorithm and the iterative retinal algorithm. The performance of the algorithm was assessed by fine-tuning the granularity of the algorithm and other parameters to optimize reconstruction accuracy and tracking efficiency.

The results demonstrate that the iterative retinal algorithm achieves a remarkable slope resolution of 0.015° and position resolution of 22.89 μm. These values are comparable to those obtained using the noniterative retinal algorithm. Moreover, the iterative retinal algorithm exhibited a tracking efficiency of over 97%. These results confirm that the retinal and iterative retinal algorithms effectively fulfill the task of tracking linear particle trajectories obtained by pixel sensors in SEE localization.

II. PRINCIPLE OF RETINAL ALGORITHM FOR LINEAR TRAJECTORY TRACKING

The retinal algorithm is inspired by the recognition characteristics of the biological retina, and is a pattern recognition algorithm invented to mimic the visual system of living organisms. Theoretically, the retinal algorithm can recognize patterns using multiple parameters. In this study, the algorithm was used to recognize straight-line trajectories projected onto a 2D plane for SEE localization.

The principle of tracking linear trajectories is as follows. First, the 2D parameter space of the straight-line patterns is divided into several small units based on the set granularity. Accordingly, a 2D parameter space array is generated. Each unit in the array is called a retinal cell in the algorithm. The corresponding parameters are denoted by (i, j). The elements from the target sample are then input into the divided retinal cell array. In each retinal cell $cell_{ij}$, the similarity weight value W_{ij_v} between the cell (i, j) and each element v is calculated. After all the elements in cell (i, j) are inserted, the W_{ij_v} from all elements belonging to the sample are accumulated as the sum result Sum_{ij} of each cell. Finally, following scanning of all the cells from the defined retinal cell array, the Sum_{ij} value of all the cells is determined. The cell with the largest value of Sum_{ij} is selected as the track, and the corresponding parameters are found by the retinal algorithm.

As inferred from the principle of the retinal algorithm, the calculation of W_{ij_v} for each computing unit in the retinal computation array can be conducted concurrently. This endows the algorithm with an extremely high parallelism, short computational latency, and the potential to serve as a real-time tracking algorithm.

The algorithm is divided into four steps as shown in Fig. 1.

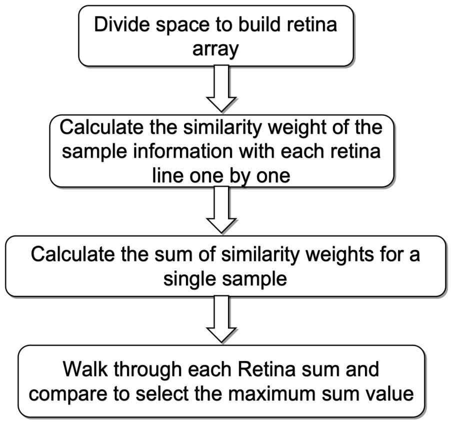


Fig. 1. Retinal algorithm computation steps.

Step 1: Partition space to establish retinal cell array. Initially, the boundary conditions were set for the large square array in both the horizontal and vertical directions of the parameter space, pre-determining the array range of values. Within this confined parameter space, the 2D space is divided into $I \times J$ small square retinal cells along both directions. Consequently, the retinal cell array comprised I rows and J columns of computational cells, known as retinal cells, with the total count defining the granularity of the algorithm. The horizontal array dimension comprises cells $R_{1j} : \{R_{11}, R_{12}, ..., R_{1j}\}$, and the vertical dimension is denoted by $R_{i1} : \{R_{11}, R_{21}, ..., R_{i1}\}$.

Step 2: Individually input sample information into the algorithm framework. For each sample, the similarity weight was calculated in relation to each retinal cell within the algorithm.

$$Sum_{ij} = \sum_{v=1}^{u} W_{ij\_v} = \sum_{v=1}^{u} e^{-\frac{d_{ij\_v}^2}{2\sigma^2}}. \quad (1)$$

Step 3: Calculate the sum of the similarity weights, denoted by Sum_{ij}, between a single case sample and each retinal cell (i, j). The formula is given in Eq. (1), where W_{ij_v} is the similarity weight value between the retinal cell at the i_{th} row and j_{th} column of the retinal cell array (referred to as retinal cell (i, j)) and the v_{th} element of the single case sample; d_{ij_v} is the distance in the vertical axis direction between the v_{th} element of the single case sample and the retinal cell (i, j); and Sum_{ij} is the total sum of similarity weights between all elements of the single case sample and the retinal cell (i, j). In addition, the variable σ is an adjustment factor for the calculation and is determined by the granularity of the retina.

Step 4: Scan all retinal cells to obtain the value distribution of Sum_{ij}. Finally, the retinal cell with the maximum value of the sum is identified as the linear particle trajectory, and the corresponding position (i, j) is referred to as the parameter reconstructed by the retinal algorithm.

To describe the use of the retinal algorithm to track the heavy-ion trajectory in 2D space, a detailed example is presented as follows.

In 2D space, a line in the Cartesian coordinate system is represented by $y = kx + b$. Under these conditions, the 2D parameter space of the retinal algorithm is defined as (k, b). The horizontal axis represents the slope k of the line predicted by the retina, with the value range set between -1 and 1, and the vertical axis represents the y-intercept b of the predicted line, with the value range set from -1 to 1. In this example, k and b are both dimensionless. The scanning granularity of the algorithm was set to $i \times j = 4 \times 4 = 16$, and the computational units in the entire parameter space were set to the granularity range from $cell_{11} \sim cell_{44}$, where $cell_{ij}$ represents the retinal cell unit in the i_{th} row and j_{th} column.

From the example, a ''slope-intercept" 2D parameter space is generated and shown on the left side of Fig. II. In the parameter space, the $cell_{23}$ and $cell_{41}$ correspond to lines in the ''x-y" 2D physical space, which are shown on the right side of Fig. II as lines L_{23} and L_{14}, respectively. The lines L_{ij} corresponding to these $cell_{ij}$ are referred to as retinal pre-matched lines, and their corresponding linear expressions are $y = jx + i$. After L_{ij} and $cell_{ij}$ are defined, the sample line trajectory is in the manner shown on the left side of Fig. II, which comprises the coordinates of the gray points. The vertical distances between these coordinate points and the retina-predicted line L_{23} are denoted by d_{23_1}, d_{23_2}, d_{23_3}, d_{23_4}. Subsequently, the sum of Eq. (1) in Section II is used to calculate the similarity weight sum for the sample in $cell_{23}$, denoted by Sum_{23}. In the same manner, all Sum values from the retinal cell array can be determined. The value distributions of the 16 sums, labeled Sum_{11} to Sum_{44}, can be obtained, as shown on the right side of Fig. II. To identify the parameters of the trajectory, all 16 sums were compared with each other, and the cell with the maximum value of the sum was selected. The coordinates (i, j) of the selected cell corresponding to the slope and intercept parameters of the particle trajectory are identified and reconstructed using an algorithm. In this step, the process of tracking and extracting reconstruction parameters for a single straight-line trajectory is completed using the retinal algorithm.

Based on the aforementioned example, a simulation was conducted to test the tracking function of the retinal algorithm. In the simulation, 2000 linear track samples were generated each time, and the ranges of values for the parameters k and b were set to [0, 10]. In addition, the granularity of the retinal algorithm was set to four different values: 10×10, 20×20, 49×49, and 99×99. The results of the slope and intercept resolutions of the line trajectories reconstructed by the retinal algorithm under the four different granularities are listed in Table 1, where σ_k represents the standard deviation for parameter k and σ_b represents the standard deviation for parameter b. As the table shows, the precision of the reconstructed parameters increases with the granularity of the retinal array. When the granularity of the algorithm was increased to 99×99, the slope and intercept resolution reached 0.03 and 0.14, respectively. These results are quite precise and indicate the feasibility of the retinal algorithm for tracking linear trajectories.

TABLE 1. Distribution of slope resolution σ_k and intercept resolution σ_b of the retinal algorithm under various granularities.

Granularity	σ_k	σ_b
10×10	0.31	1.38
20×20	0.16	0.7
49×49	0.06	0.27
99×99	0.03	0.14

After obtaining preliminary results from the above simulation, two types of tracking algorithms with fixed parameters were designed for track fitting with heavy iron particles. One is a simple retina for tracking a single-track event, and the other is an iterative retina for tracking multiple-track events. The two algorithms are described in detail in the following section.

III. RETINAL ALGORITHM IMPLEMENTATION FOR SINGLE-PARTICLE TRACK FITTING

This section focuses on the experimental study of single-particle trajectory tracking. Building on the foundational steps of the retinal algorithm, two novel parameters were incorporated: a distance threshold and a pixel value weight. These additions aimed to provide a more comprehensive assessment of the tracking accuracy achieved using the retinal algorithm. For the analysis, single-track sample data derived from a 5000-frame simulation via the Topmetal sensor were utilized as the input for the algorithm.

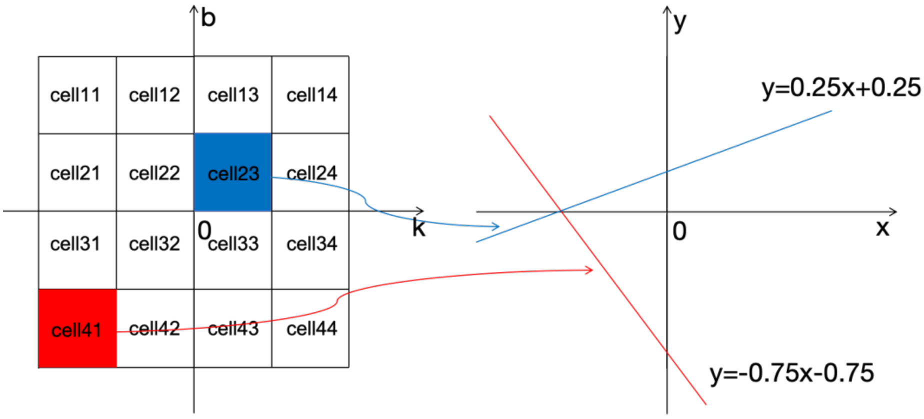


Fig. 2. Retinal parameter space (left) and corresponding pre-matched line in the x-z plane (right).

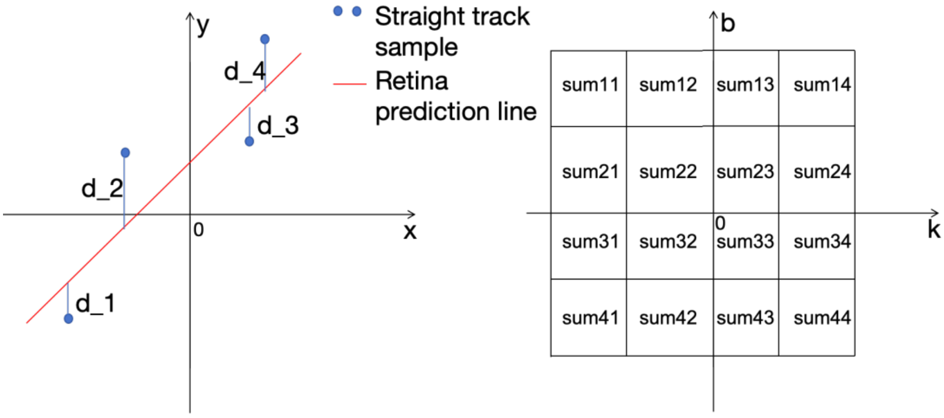


Fig. 3. Coordinate positions of the sample and retinal prediction line (left), and sum distribution of the sample in algorithm granularity of 4 × 4 (right).

A. Single-track event samples

When charged particles pass through a gas detector, a certain number of electron-ion pairs are generated along the particle track. Electrons and ions drift in opposite directions under the influence of an electric field. The electrons are then collected by the pixel array of a Topmetal sensor, which serves as the anode, thereby converting the generated electrical signal into a track image of the incident particles. When placed on a single plane, the Topmetal sensor is capable of providing one-dimension (1D) position information of incident particles. When two Topmetal sensors are strategically placed on two orthogonal planes, 2D position information of incident particles can be acquired.

The principle behind using a Topmetal sensor to locate a single heavy ion is illustrated in the upper part of Fig. 4. The positioning module of the tracking detection device consists of two identical positioning units, each of which has a flat cathode and a silicon pixel sensor that acts as both an anode and for track imaging. A 3D schematic of the device and the internal structure of the unit are shown on the left side of the lower part of Fig. 4. The device under test (DUT) for SEL testing was placed on the right side of the second unit. To minimize the edge effects of the electric field, a cage structure composed of thin metal strips was added to both sides of the electrodes. This addition aims to reduce electric field distortion at the edges. Furthermore, the pixel sensor was placed away from the boundaries of the unit to minimize the impact of electric field edge effects as much as possible. When high-energy heavy ions pass through the units, the electrons produced along the track drift towards the pixel array under the influence of the electric field. During the drift process, electrons diffuse, ultimately forming a straight projected track of heavy ions on the pixel sensor. The pixels with high brightness (yellow pixels) represent the track of the heavy ion projected onto the x-z plane, as shown on the right side of the

lower part of Fig. 4. The position between the positioning unit and the DUT was carefully calibrated before testing. One positioning unit can provide 1D positional information for a single heavy ion; therefore, two units can provide 2D coordinates of the heavy ion hitting the DUT. During the track simulation, 25 MeV/u Kr ions were used as incident particles with a gas composition of $Ar : CO_2$(70:30) at 0.8 atm and 300 K. The electric field strength was 300 V/cm. The other detailed parameters are listed in Table 2. The pixel array of the Topmetal sensor was 72 × 72, with a pixel pitch of 83 μm, and the effective area of the pixel-sensitive region was 6 mm × 6 mm. During the fitting process, pixels with analog values greater than 10 were input into the algorithm to obtain the position information of the heavy ion.

B. Retinal algorithm parameter design

Based on the principle of the retinal algorithm introduced in Section II and combining the features of the particle linear trajectory samples, the basic parameters of the retinal algorithm can be determined. The 2D parameter space of the retinal algorithm is (k, b). Here, k represents the slope of the line, and the value range is set between -1 and 1. b represents the y-intercept of the line, with values ranging from 0 to 72 times the pixel pitch. To conduct a detailed evaluation of the algorithm, multiple levels of granularity were set for the retinal algorithm, ranging from 31 × 31 to 95 × 95.

To enhance the computational accuracy of the retinal algorithm, three new influencing factors were next incorporated in the process of calculating the weight value W of the retinal cell. These factors were the pixel value, pixel threshold, and distance threshold.

First, based on the existing formula for calculating the weight value W, the pixel value A_n of each input sample coordinate point n is introduced as an influencing factor. This implies that the pixel value of each coordinate point in the input data contributes to the calculation of the weight value W, which is likely to be a factor determining the importance or influence of that particular point in the processing of the algorithm. The pixel value An of each input sample's coordinate point n was first incorporated as an influencing factor into the original formula to calculate the weight value W. With this change, the formulas for calculating W and Sum are expressed as Eq. (2). This resulted in pixels with higher brightness in the trajectory sample having a higher matching weight for the predicted line L, represented by the corresponding cell. This improved the tracking accuracy of the retinal algorithm.

$$Sum_{ij} = \sum_{n=1}^{m} W_{ij\_k} = \sum_{n=1}^{m} A_n \cdot e^{-\frac{d_{ij\_n}^2}{2\sigma^2}}. \tag{2}$$

Second, two parameters, the distance threshold (denoted as $d_{threshold}$ or d_{th}) and pixel threshold (denoted as $A_{threshold}$ or A_{th}), were introduced into the algorithm, and the calculation formula was modified from Eq. (2) into Eq. (3). Given that a sample of a straight-line trajectory contains a large number of valid coordinate points, the introduction of these two thresholds, d_{th} and A_{th}, enables the algorithm to disregard the coordinate positions that are significantly far from the predicted line L ($d_{ij\_n} > d_{th}$) or have a low pixel value (A_n). This improvement not only reduces the computational burden but also enhances the tracking efficiency and accuracy of the algorithm.

$$\begin{aligned} Sum_{ij} &= \sum_{n=1}^{m} W_{ij\_n} \\ &= \sum_{n=1}^{m} A_n \cdot \begin{cases} e^{-\frac{d_{ij\_v}^2}{2\sigma^2}}, d_{ij\_n} \le d_{th} \wedge A_n \ge A_{th}. \\ 0, d_{ij\_n} > d_{th} \vee A_n < A_{th}. \end{cases} \end{aligned} \tag{3}$$

Third, based on the heavy-ion trajectory samples, the pixel threshold value A_{th} is set to 10 brightness levels, and the distance threshold d_{th} is set to 1.5 × the bin size of the retinal cell along the intercept parameter, which is equal to (1.5 × 83 × 72 / J) μm, where J is the granularity of the retinal algorithm along the intercept parameter.

C. Performance results of tracking of single-track event samples

After setting the parameters of the retinal algorithm, a complete simulation test was conducted to evaluate its ability to track single-trajectory samples. Three parameters were used as criteria for measuring the accuracy of the algorithm: slope resolution, intercept resolution, and comprehensive position resolution of the reconstruction.

For single-track sample data, each frame contains one track, assuming that the number of frames obtained from the Topmetal sensor is N, and the slopes of the N original straight-line samples are $[k_1, k_2, k_3, ..., k_N]$. If the slopes of each trajectory reconstructed by the retinal algorithm are $[K_1, K_2, K_3, ..., K_N]$, the slope resolution σ_k can be expressed as

$$\sigma_k = sqrt\left[\frac{(k_1 - K_1)^2 + (k_2 - K_2)^2 + ... + (k_N - K_N)^2}{N-1}\right]. \tag{4}$$

Similarly, the reconstruction resolution of the longitudinal intercept, denoted σ_b, is defined as

$$\sigma_b = sqrt\left[\frac{(b_1 - B_1)^2 + (b_2 - B_2)^2 + ... + (b_N - B_N)^2}{N-1}\right], \tag{5}$$

where $[b_1, b_2, b_3, ..., b_N]$ represents the longitudinal intercepts of the N original line samples, and $[B_1, B_2, B_3, ..., B_N]$ corresponds to the longitudinal intercept parameters of each track reconstructed by the retinal algorithm.

$$P_{res} = sqrt\left[\frac{P_1^2 + P_2^2 + ... + P_N^2}{N-1}\right]. \tag{6}$$

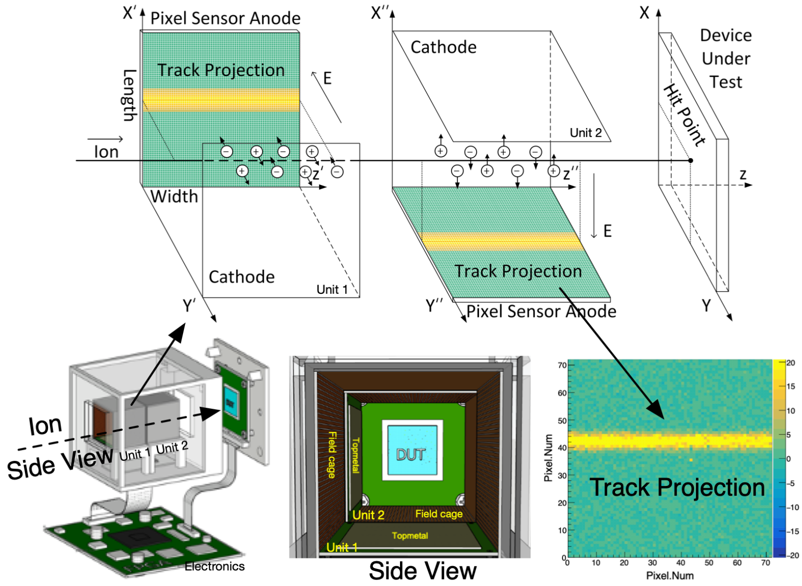


Fig. 4. Principle and schematic of the heavy-ion tracking detection device based on a Topmetal sensor and the track projection of a single heavy ion.

TABLE 2. Various conditions for the formation of a single track on the pixel detector.

Parameter	Value	Parameter	Value
Particle types	Kr^{26+}	Height of incidence (mm)	1.5~5
Particle energy (MeV/u)	25	Electric field (V/cm)	300
Gas	$Ar : CO_2$(70:30)	Drift speed (cm/μs)	0.7
Gas density (mg/cm^3)	1.714	Diffusion coefficient ($\mu m/\sqrt{cm}$)	150
Ionized energy W_I (eV)	28.4	Pixel pitch (μm)	83
Energy loss (keV/μm)	3.06	pixel array	72×72
Range of ions in gas (mm)	513		

$$P = \frac{\sum_{k=1}^{72} |y_k - Y_k|}{72}. \tag{7}$$

Position resolution is commonly employed as a key metric in the assessment of the tracking performance for straight-line trajectories. Based on the samples and algorithms used, the calculation method for the position resolution (P_{res}) is expressed in Eq. (6). where P_N is the trajectory reconstruction position deviation obtained from the N_{th} sample, and its calculation formula is shown in Eq. (7). Consider a scenario in which the original 2D parameters of a linear trajectory in the sample are represented by (k, b), and the 2D parameters reconstructed by the algorithm are (K, B). Consequently, the original linear trajectory is described by $y = kx + b$, whereas the reconstructed linear trajectory follows $y = Kx + B$. When the variable x assumes values ranging from 0 to 71 times the pixel size, i.e. [0, 1 $\times$ 83 μm, 2 $\times$ 83 μm, ..., 71 $\times$ 83 μm], the resultant values for the original and reconstructed tracks can be denoted as $[y_1, y_2, y_3, \ ... \ , y_{71}, y_{72}]$ and $[Y_1, Y_2, Y_3, \ ... \ , Y_{71}, Y_{72}]$, respectively.

During the process of reconstructing the single-track samples, we explored the impact of varying the granularity parameters on the reconstruction accuracy of the retinal algorithm. This relationship is illustrated in Fig. 5. Fig. 5(a), and Fig. 5(b) presents the detailed distributions of the algorithm's reconstruction resolution of the slope, resolution of the intercept, and position resolution at different granularities. The results indicate that, as the granularity of the algorithm increases, there is a corresponding enhancement in the tracking accuracy, which is particularly noticeable in the improvement of the intercept and position resolutions. However, beyond a certain granularity level, further increases do not significantly enhance tracking accuracy, which tends to stabilize at a certain value. At this point, the accuracy of the reconstructed linear trajectory predominantly depends on the precision of the trajectory samples and the intrinsic properties of the detector, aligning with the theoretical expectations of the algorithm.

In practical applications of the algorithm, it is crucial to strike a balance between granularity and accuracy. The goal was to minimize the algorithm granularity to reduce the computational load while still satisfying the resolution required for recognition. For the single-track reconstruction detailed in this study, a set of algorithm parameters was optimized, as listed in Table 3. Specifically, the variance σ was set to 0.8, the distance threshold was set to 0.4 pixels, and granularity was set to 72 $\times$ 100. The resulting calculations yielded an an-

TABLE 3. Parameter Settings in the Algorithm and Distribution of Experimental Results.

Granularity	σ	A_{th}	Number of Samples	d_{th}	σ_k	σ_b	Position resolution
72×72	0.8	10 pixels	5500	0.4 pixel	$0.573°$	25 μm	23μm

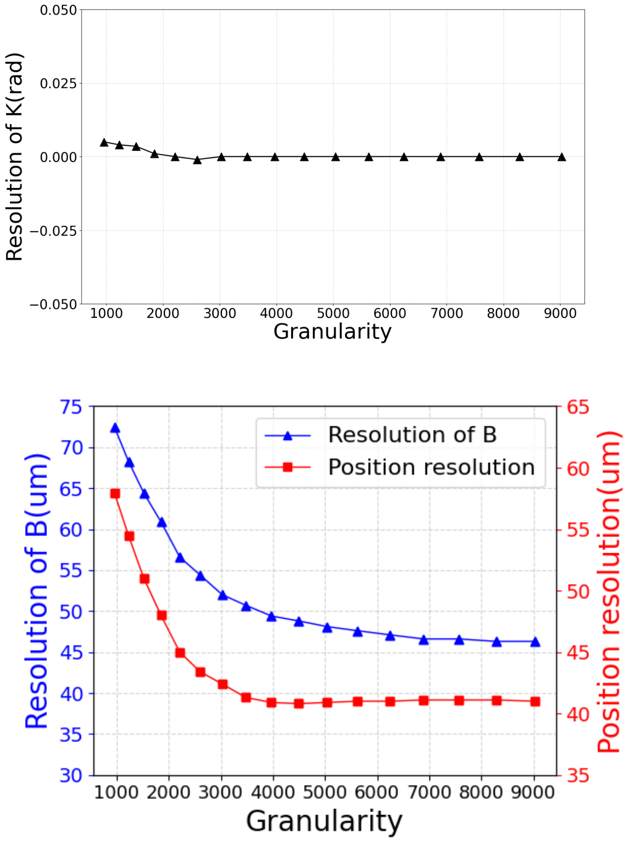


Fig. 5. Distribution of slope, intercept, and position resolution of tracking at different granularities.

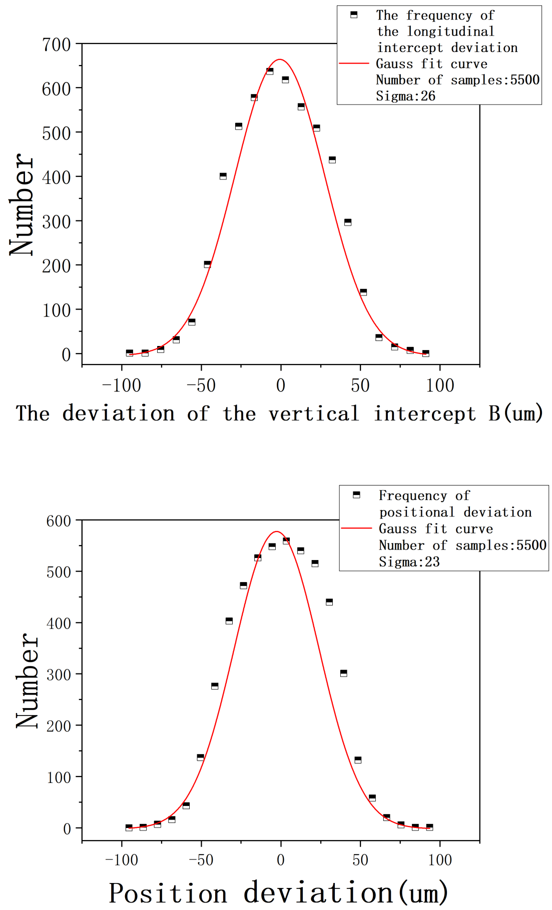


Fig. 6. Distribution of the deviation of the vertical intercept b (left) and position (right) counting 5500 single-track event samples in the case of retina scanning granularity of 72×100.

gular resolution σ_k of 0.573° and longitudinal resolution σ_b of 25 μm, as shown in Fig. 6(a). These results demonstrate that the algorithm maintains an angular error within 0.6° and a longitudinal error within 1/3 of the pixel pitch (83 μm), thereby confirming its accuracy in track reconstruction.

Furthermore, by applying the retinal algorithm with the established parameters, a distribution graph illustrating the position resolution was obtained from 5500 single-track samples, as depicted in Fig. 6(b). The solid black line in the figure illustrates the actual distribution of the resolution deviation, and the red curve was obtained via Gaussian fitting. This distribution reveals that, under these conditions, the final positional resolution of the retinal algorithm is approximately 23 μm, which is notably less than 50% of the pixel pitch (83 μm). This resolution level indicates that the algorithm preliminarily meets the resolution requirements of Topmetal sensors for tracking the linear trajectories of heavy ions and is expected to achieve a higher positional resolution at higher pixel densities.

IV. IMPLEMENTATION OF ITERATIVE RETINAL ALGORITHM FOR MULTITRACK EVENT FITTING

The retinal algorithm performs well by reconstructing the trajectories generated from single-track event samples. The retinal algorithm needs to evolve to handle multiple event samples. To meet the practical requirements, an iterative retina (IR) algorithm was proposed and applied to track multiple trajectories. Samples based on multiple trajectory events (referred to as multitrack events) were used as inputs for the IR algorithm. The efficiency and resolution of the IR algorithm are discussed at the end of this section.

A. Multitrack event samples

Under the same conditions and environment as when generating single-track event samples, multiple Kr ion beams were inserted during multitrack event generation. In this manner, the Topmetal sensor can obtain data samples formed by multiple straight-line trajectories each time, as shown in Fig. 7. These data samples were multitrack event samples used as the input for the IR algorithm. The simulation parameters of the multitrack event samples were the same as those of the single-track event samples in Section III A. For multitrack positioning, the beam should be uniformly distributed in the cross-section, and the beam size can be adjusted to cover the area of the DUT by a collimating orifice, while the beam width should be smaller than the detectable width of the positioning module. The experiment showed that the maximum beam flux of the Kr ions was 10^3 ions/$(cm^2 \cdot s)$ under room conditions, corresponding to approximately five tracks in each frame. At higher fluxes, the tracks overlap significantly and cannot be distinguished effectively [37]. Considering the pixel size of the Topmetal sensor, during the event generation process, the number of incident Kr heavy ions in a single-event sample was set between three and seven.

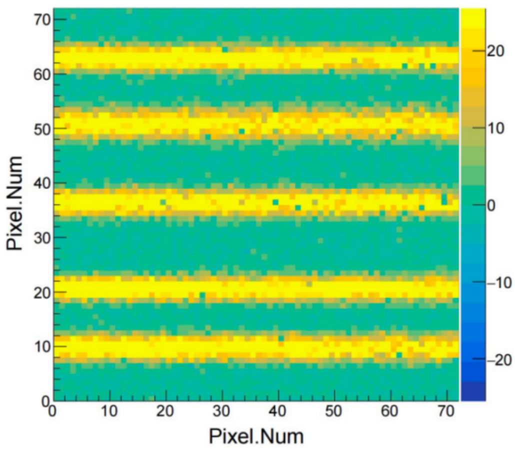


Fig. 7. Pixel distribution graph of a multitrack event sample in the x-z plane.

B. Principle of iterative retinal algorithm

Compared with the single-track event tracking case, in the multitrack event case, all particle trajectories present in the event must be identified simultaneously. This requires the algorithm to determine the ability to find maximum sum values. According to the principle of the retinal algorithm, only one retinal cell with the maximum sum value in a given 2D parameter space is found each time. Therefore, an additional IR algorithm was designed to satisfy the requirement of determining the maximum sum values in a given region.

The main concept of the IR algorithm is to run the retinal algorithm iteratively. Consider the two-iteration retinal algorithm as an example (shown in Fig. 8). In the first iteration, the full parameter space is divided into $I \times J$ retinal computing units, and these computing units are referred to as super retinal cells (supercells). Scanning using the first retinal algorithm enables the sum of the samples to be obtained for each supercell. Supercells with sum values exceeding the threshold Sum_{th1} (represented by blue squares in Fig. 8) are selected and entered into the second round of the retinal algorithm for iterative calculations. In the second iteration, the parameter space contained in each selected supercell is further divided into $Q \times Q$ retinal computing units, which are regarded as ordinary retinal cells. After the second round of scanning, the retinal cell with the maximum sum value from the selected supercell is selected. The sum values with the threshold Sum_{th2} are then compared. If the sum exceeds the threshold Sum_{th2}, a pattern is successfully identified in the cell space, as indicated by the red squares in the cell area in Fig. 8. If it does not exceed the threshold Sum_{th2}, no valid pattern is matched to that supercell. The values of Sum_{th1} and Sum_{th2} are related to the number of elements in each input sample.

This study employs a second-order iterative retinal algorithm for tracking trajectories in multitrack event samples. The granularity is set to $Z \times Z$ in the first iteration and $Q \times Q$ in the second iteration. The settings for the pixel and distance thresholds in both iterations aligned with the rules established for the ordinary retinal algorithm, as described in the previous section. Furthermore, two additional filtering thresholds, Sum_{th1} and Sum_{th2}, were implemented to identify the retinal cell units where the maximum values were located, thereby facilitating the completion of the tracking task. The algorithmic process designed for this purpose is illustrated in Fig. 9.

In addition, compared to the simple retinal algorithm, the IR algorithm can reduce the computational load of the entire algorithm when the scanning granularity is the same. Assuming that the IR algorithm selects H supercell units in the first iteration, the total number of retinal computation units that must be calculated is $(Z \times Z + H \times Q \times Q)$. For the simple retinal algorithm, to achieve the same granularity of the scanning computation, we need to compute $(Z^2 \times Q^2)$ retinal cells. Therefore, the computational load of the IR algorithm is $(1/Z^2 + H/Q^2)$ times that of the conventional retinal algorithm.

C. Efficiency calculation of IR track algorithm

In addition to the resolution of the reconstruction parameters, the efficiency of the algorithm is an important factor in determining its performance in the case of multitrack event samples. According to the trajectory results identified by the IR algorithm, the matching patterns of the identified trajectories can be classified into three main categories, defined as follows.

- Perfectly matched trajectory: The trajectory identified by the algorithm is uniquely identical to the trajectory in the multitrack event samples.

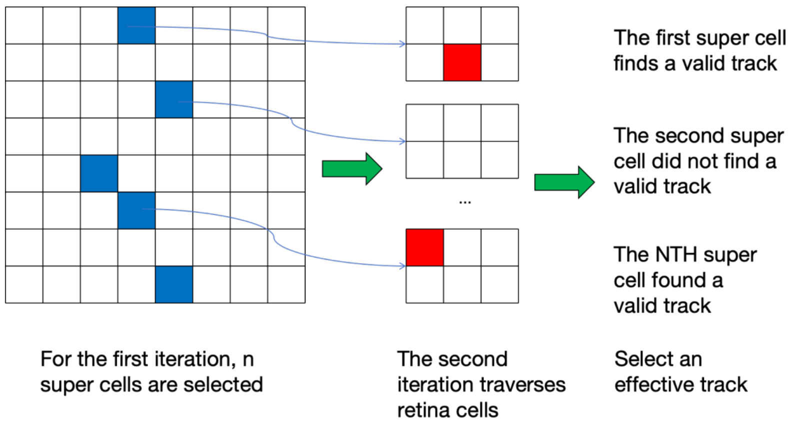


Fig. 8. Steps of the two-iteration retinal algorithm.

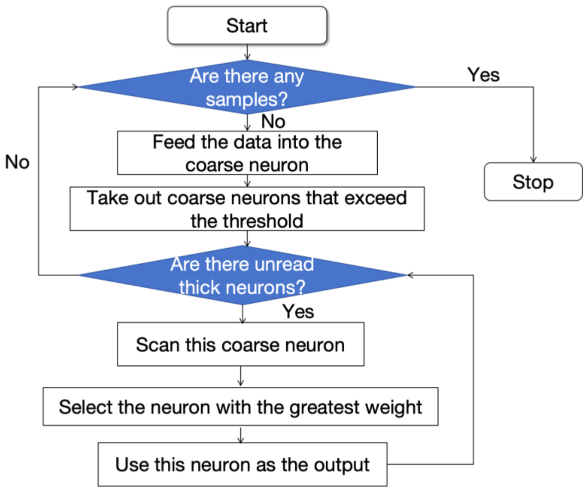


Fig. 9. Work flow of the IR algorithm.

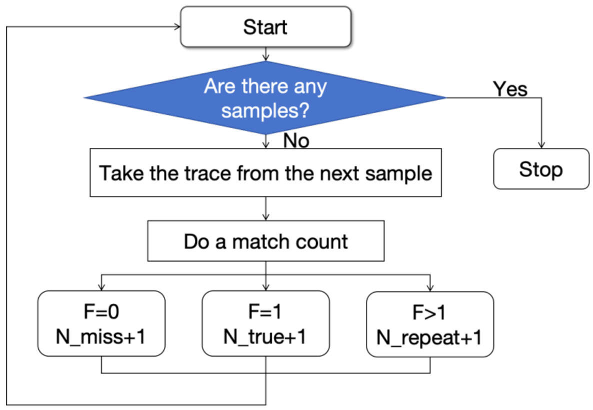


Fig. 10. Flow process of the perfectly matched/over-matched/unmatched trajectory ($N_{repeat}/N_{miss}/N_{true}$) counting.

- Over-matched trajectory: Identified trajectories are the same as those in the multitrack event samples, but more than one exists.
- Unmatched trajectory: A trajectory that exists in the multitrack event samples but is not present in the identified results of the IR algorithm.

The trajectories belonging to the first two cases are valid tracks reconstructed by the IR algorithm, whereas the trajectories belonging to the third case are missing tracks that the IR algorithm does not find. For example, the IR algorithm can be used to trace N multi-trajectory samples and identify M heavy-iron trajectories. In the resulting analysis, these M trajectories must be classified individually according to the matching pattern defined above. The classification and statistical processes for the three matching patterns ($N_{repeat}/N_{miss}/N_{true}$) are shown in Fig. 10. In this process, the existing trajectories from the multitrack event sample are taken as input and sequentially compared with all the trajectories found by the IR algorithm. The number of identical coordinate points (denoted as Num_s) between each existing trajectory and the trajectory identified by the IR algorithm is counted, and a judgment is made based on the value of Num_s. This process determines whether the two trajectories match. Each successful match increases the match count F by one. After inputting and analyzing all original trajectory samples, the matching count F, where F equals 1, represents a perfectly matched trajectory. However, a matching count F greater than 1 for a straight trajectory indicates an over-matched trajectory, whereas a matching count F equal to zero represents an unmatched trajectory.

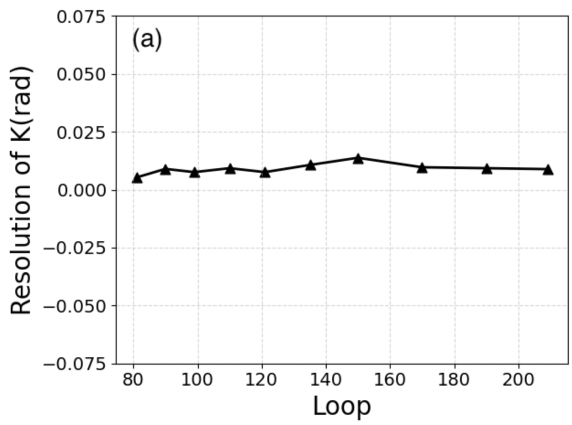
$$efficiency = \frac{N_{all} - N_{miss}}{N_{all}} \times 100\%. \tag{8}$$

$$N_{all} = N_{true} + N_{repeat} + N_{miss}. \tag{9}$$

After statistical results are obtained for the three types of matched trajectories, namely, N_{repeat}, N_{miss}, and N_{true}, in the IR algorithm, the efficiency of the algorithm can be calculated using Eq. (8), where N_{all} denotes the total number of trajectories originally present in the sample. The relationship between N_{all} and $N_{true}/N_{repeat}/N_{miss}$ is given by Eq. (9) as follows:

D. Performance of the IR algorithm for tracking multitrack event samples

The performance of the IR algorithm was tested according to the methods introduced in the previous subsections. During testing, IR algorithms with different granularities were configured to comprehensively evaluate the performance of the algorithm. The results also provide detailed parameter references for implementing the IR algorithm in real-time FPGA devices. The tracking results of the IR algorithm are presented in the following figures.



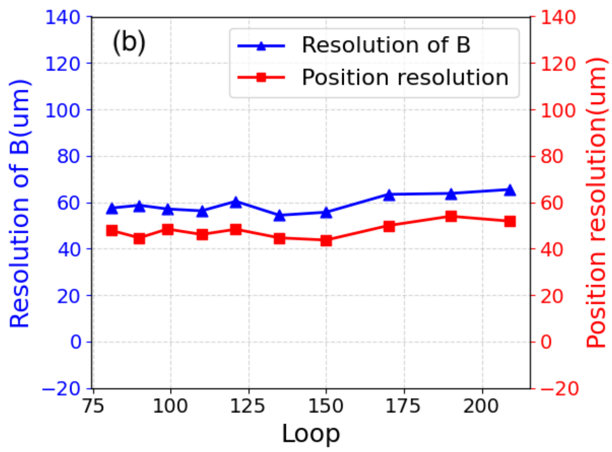


Fig. 11. Results of the reconstruction resolution for slope k (left), vertical intercept b, and position resolution (right) in the case of different granularities of the IR algorithm.

Resolution results for the reconstructed parameters are shown in Fig. 11. Fig. 11(a) and Fig. 11(b) show the distribution of the reconstruction resolutions of the slope, intercept, and position under different granularities of the IR algorithm. In the figure, a parameter loop is used instead of the granularity of the two-iteration retinal algorithm. The relationship between the loop and the granularity of the two iterations is given by Loop = $Z \times Q$, where the granularity of the first iteration of the retinal algorithm is $Z \times Z$, and the granularity of the second iteration is $Q \times Q$. Ten different granularities were set from loop = $Z \times Q = 8 \times 10$ to loop = $Z \times Q = 19 \times 11$ in the IR retinal result analysis. As the graph shows, under different granularity situations, the reconstruction parameter resolution of the algorithm remains stable. This differs from the case of using the retina in a single-track event, where the smallest granularity (loop = $Z \times Q = 8 \times 10$) in the IR algorithm is already sufficiently high to reconstruct tracks accurately. Therefore, with other parameters set reasonably, the IR algorithm can accurately match the positions of the lines, achieving a high level of tracking accuracy that meets the basic requirements for tracking.

In addition, compared with the single-track event case, the position resolution in the multitrack event case is degraded from 40 to 50 μm on average under the same granularity. This is because, compared to the retinal algorithm, in the case of the IR algorithm, we appropriately lowered the threshold-setting standard during the two iteration scans to maintain the efficiency of the IR algorithm at over 90%. Approximately 8% of the overmatched trajectories were generated by the IR algorithm. The overmatched trajectories have a lower position resolution than the perfectly matched trajectories; they reduce the overall position resolution of the IR algorithm. In our future study, an additional judgment process will be added after the IR algorithm to remove those overmatched trajectories and turn them into perfectly matched trajectories.

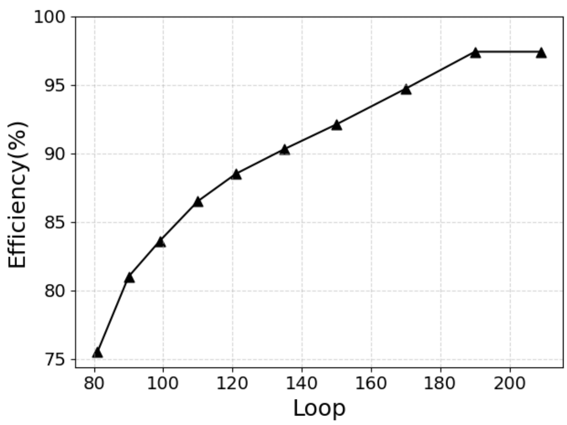


Fig. 12. Efficiency result in case of different granularities of the IR algorithm.

The tracking efficiency of the IR algorithm is presented in Fig. 12. The figure shows that a finer granularity can significantly improve the tracking efficiency of the IR algorithm. The main reason for this is that in the case of lower granularity, the parameter space covered by each supercell in the first iteration of the algorithm is too large. This significantly increases the probability of two or more original lines from different samples falling into the same supercell space. After the second calculation, the IR algorithm will only find one of these original lines, leading to the omission of the others and resulting in lower efficiency. When the granularity loop is greater than 140, the efficiency of the IR algorithm can exceed 90%, making the algorithm reliable and suitable for multitrack event sample tracking. In addition, due to the

multi-event sample used in this study, no crossing cases occurred between the trajectories in the 72 × 72 silicon pixel detector space. During the simulation, we found that the algorithm recognized only fake trajectories when there were intersections between the trajectories. Under these conditions, the IR algorithm provided a purity of 100%. All tracks found by IR can be matched with the tracks existing in the multi-event sample.

TABLE 4. Performance results of the IR algorithm under the granularity of loop = 19 × 10.

Parameter	Value	Parameter	Value
Z × Z	19 × 19	N_{all}	30000
Q × Q	10 × 10	N_{true}	26806
σ	9.5	N_{repeat}	2415
σ_I	0.4	N_{miss}	779
Pixel threshold	10 pixel	Efficiency	97.4%
d_threshold	3.4 pixel	Variance k	0.009 rad
d_threshold_I	0.25 pixel	Variance b	63 μm
Position resolution	53 μm		

Finally, the optimal combination of granularity and parameters was determined. A total of 30,000 straight-line reconstructed trajectories obtained from 5,000 multiparticle trajectory samples were counted and analyzed. The tracking performance of the IR algorithm for this combination is listed in Table 4. From the table, the granularity of the IR algorithm was set to loop = $Z \times Q$ = 10 × 19, and the efficiency of the IR algorithm reached over 97% with a position resolution of 53 μm. The performance results indicate that the IR algorithm presents a high tracking efficiency for multitrack heavy-ion samples while providing acceptable reconstruction accuracy. This meets the requirements for SEE located in 2D space on the Topmetal pixel detector platform.

V. CONCLUSIONS

In this study, a tracking algorithm based on the retinal algorithm was developed to recognize and reconstruct linear trajectories in 2D space for SEE localization. For the gas detector, a Topmetal sensor with a 72 × 72 pixel array and a pixel pitch of 83 μm was selected to simulate single and multiple linear track event samples of 25 MeV/u heavy ions under the gas condition of $Ar : CO_2$ (70:30) at 0.8 atm and 300 K. The retinal algorithm then processed single-track samples, whereas the iterative retinal algorithm handled multi-track events. The simulation results showed that the algorithm achieved a track position resolution of 40 μm for both single- and multitrack events. In addition, the iterative algorithm achieved a tracking efficiency exceeding 97%. These results indicate that the retinal algorithm is highly effective at tracking the linear trajectories of heavy ions using a pixel detector. Furthermore, its high parallel-scanning computing capability is promising for online tracking applications. Efforts are being made to implement the retinal algorithm in FPGA devices to achieve rapid online track fitting for SEE localization, targeting processing times of 5 $\sim$ 8 μs per event.

[1] Pearton S J, Aitkaliyeva A, Xian M, et al. Review—Radiation Damage in Wide and Ultra-Wide Bandgap Semiconductors[J]. ECS Journal of Solid State Science and Technology, 2021, 10(5): 055008.doi:10.1149/2162-8777/abfc23

[2] Höeffgen S K, Metzger S, Steffens M. Investigating the effects of cosmic rays on space electronics[J]. Frontiers in Physics, 2020, 8: 318.doi:10.3389/fphy.2020.00318

[3] C. Dyer, A. Hands, K. Ryden, et al. Extreme atmospheric radiation environments and single event effects[J]. IEEE Transactions on Nuclear Science, 2017, 65(1): 432-438. doi:10.1109/TNS.2017.2761258

[4] Zhao, Y. F., Wang, L., Yue, S. G, et al. Single event effect and its hardening technique in nano-scale CMOS integrated circuits[J]. Acta Electron. Sin, 2018, 46(10): 2511-2518. doi:10.3969/j.issn.0372-2112.2018.10.027

[5] R.L. Pease, Total ionizing dose effects in bipolar devices and circuits[J]. IEEE Transactions on Nuclear Science, 2003, 50(3): 539-551. doi:10.1109/TNS.2003.813133

[6] J.R. Srour, J.W. Palko, Displacement damage effects in irradiated semiconductor devices[J]. IEEE Transactions on Nuclear Science, 2013, 60(3): 1740-1766. doi:10.1109/TNS.2013.2261316

[7] Shengli Chen, Assessment of neutron irradiation effects via PKA spectra, displacement damage, and gas production: Application to reactor pressure vessel. Nuclear Materials and Energy. **37**, 101512 (2023), doi:10.1016/j.nme.2023.101512

[8] F.W. Sexton, Destructive single-event effects in semiconductor devices and ICs[J]. IEEE Transactions on Nuclear Science, 2003, 50(3): 603-621. doi:10.1109/TNS.2003.813137

[9] Lopez-Calle, I., Franco, A.I. Comparison of cubesat and microsat catastrophic failures in function of radiation and debris impact risk. Sci Rep 13, 385 (2023). doi:10.1038/s41598-022-27327-z

[10] Nwankwo, V. U., Nnamdi N. Jibiri, and Michael T. Kio. The impact of space radiation environment on satellites operation in near-Earth space[J]. Satellites Missions and Technologies for Geosciences, 2020: 73-90.doi:10.5772/intechopen.90115

[11] Yu Q, Sun Y, Luo L, et al. Experimental study of Single Event Upsets on deep submicron and nano devices in space[C]. 2017 Prognostics and System Health Management Conference (PHM-Harbin). IEEE, 2017: 1-4.doi:10.1109/PHM.2017.8079299

[12] Sehra K, Kumari V, Gupta M, et al. Impact of heavy ion particle strike induced single event transients on conventional and π-Gate AlGaN/GaN HEMTs[J]. Semiconductor Science and Technology, 2021, 36(3): 035009.doi:10.1088/1361-6641/abdba3

[13] Xin J, Zhu X, Ma Y, et al. Study of Single Event Latch-Up Hardness for CMOS Devices with a Resistor in Front of DC-DC Converter[J]. Electronics, 2023, 12(3): 550.doi:10.3390/electronics12030550

[14] Peng C, Lei Z, Chen Z, et al. Experimental and simulation studies of radiation-induced single event burnout in SiC-based power MOSFETs[J]. IET Power Electronics, 2021, 14(9): 1700-1712.doi:10.1049/pel2.12147

[15] Qiming Chen, Gang Guo, Li Sui, Jiancheng Liu, Yanwen Zhang, Fuqiang Zhang, Qian Yin, Jinhua Han, Zheng Zhang, Haohan Sun. Accelerator simulation test technology and its application for single event effect evaluation in space[J]. NUCLEAR TECHNIQUES, 2023, 46(8): 080008doi:10.11889/j.0253-3219.2023.hjs.46.080008

[16] Buchner, S. P., Miller, F., Pouget, V., McMorrow, D. P. (2013). Pulsed-Laser Testing for Single-Event Effects Investigations. IEEE Transactions on Nuclear Science, 60(3), 1852–1875. doi:10.1109/tns.2013.2255312

[17] Guo J, Du G, Bi J, et al. Development of single-event-effects analysis system at the IMP microbeam facility[J]. Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 2017, 404: 250-253.doi:10.1016/j.nimb.2017.02.012

[18] He Y, Zhao J, Shu W, et al. Radiation-Induced Failures for Integrated Circuits in Space and Design Philosophy[C], 2022 IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA). IEEE, 2022: 1-4.doi:10.1109/IPFA55383.2022.9915726

[19] A. Strandlie, Track reconstruction—from bubble chambers to the LHC[J]. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2004, 535(1-2): 57-64. 10.1016/j.nima.2004.07.112

[20] Nicotra D, Martinez M L, de Vries J A, et al. A quantum algorithm for track reconstruction in the LHCb vertex detector[J]. Journal of Instrumentation, 2023, 18(11): P11028. doi:10.1088/1748-0221/18/11/P11028

[21] Aad, G., Abbott, B., Abeling, K. et al. Performance of the reconstruction of large impact parameter tracks in the inner detector of ATLAS. Eur. Phys. J. C 83, 1081 (2023). doi:10.1140/epjc/s10052-023-12024-6

[22] The CMS collaboration et al. Electron and photon reconstruction and identification with the CMS experiment at the CERN LHC[J]. Journal of Instrumentation, 2021, 16(05): P05014. doi:10.1088/1748-0221/16/05/P05014

[23] Günther P A. LHCb's Forward Tracking algorithm for the Run 3 CPU-based online track-reconstruction sequence[J]. arXiv preprint arXiv:2207.12965, 2022. doi:10.48550/arXiv.2207.12965

[24] Hennequin A, Couturier B, Gligorov V V, et al. A fast and efficient SIMD track reconstruction algorithm for the LHCb Upgrade 1 VELO-PIX detector[J]. Journal of Instrumentation, 2020, 15(06): P06018. doi:10.1088/1748-0221/15/06/P06018

[25] Amrouche, S. et al. (2020). The Tracking Machine Learning Challenge: Accuracy Phase. In: Escalera, S., Herbrich, R. (eds) The NeurIPS '18 Competition. The Springer Series on Challenges in Machine Learning. Springer, Cham. doi:10.1007/978-3-030-29135-8_9

[26] Hatfield, P.W., Gaffney, J.A., Anderson, G.J. et al. The data-driven future of high-energy-density physics. Nature 593, 351–361 (2021). doi:10.1038/s41586-021-03382-w

[27] T. Boccali, High Energy Physics computing for the next decade[J]. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2022, 1043: 167434. 10.1016/j.nima.2022.167434

[28] He, WB., Ma, YG., Pang, LG. et al. High-energy nuclear physics meets machine learning. NUCL SCI TECH 34, 88 (2023). doi:10.1007/s41365-023-01233-z

[29] Ai, P., Wang, D., Sun, X., Huang, G., & Li, Z. (2020). A deep learning approach to multi-track location and orientation in gaseous drift chambers. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 984, 164640. doi:10.1016/j.nima.2020.164640

[30] Lü, L., Yi, H., Xiao, Z. et al. Conceptual design of the HIRFL-CSR external-target experiment. Sci. China Phys. Mech. Astron. 60, 012021 (2017). doi:10.1007/s11433-016-0342-x

[31] H.L. Wang, Z. Wang, C.S. Gao, et al. Design and tests of the prototype beam monitor of the CSR external target experiment[J]. Nuclear Science and Techniques, 2022, 33(3): 36. doi:10.1007/s41365-022-01021-1

[32] Hu, Y.-X.; Yang, H.-B.; Zhang, H.-L.; Liao, J.-W.; Mai, F.-T.; Zhao, C.-X. An Online Fast Multi-Track Locating Algorithm for High-Resolution Single-Event Effect Test Platform. NUCL SCI TECH 2023, 34 (5). doi:10.1007/s41365-023-01222-2

[33] Borisyak, M., Ustyuzhanin, A., Derkach, D., Belous, M., Numerical optimization for Artificial Retina Algorithm. Journal of Physics: Conference Series, 898(3). (2017), doi:10.1088/1742-6596/898/3/032046

[34] Deng, W., Lentdecker, G. DE., Huang, G., Robert, F., Song, Z., Wang, D., Yang, Y., Study of track reconstruction using Retina algorithm for charged particles in magnetic field. Proceedings of Science, 343(2018).

[35] Z.X. Song, W.D.Deng, De Lentdecker G, et al. Study of the retina algorithm on FPGA for fast tracking[J]. Nuclear Science and Techniques, 2019, 30(8): 127. doi:10.1007/s41365-019-0643-x

[36] W.D. Deng, Z.X. Song, G.M. Huang, et al. Iterative Retina for High Track Multiplicity in a Barrel-Shaped Tracker and High Magnetic Field[J]. IEEE Transactions on Nuclear Science, 2021, 68(8): 1937-1943.doi:10.1109/TNS.2021.3086272

[37] Zili Li, Yan Fan, Zhen Wang, Jun Liu, Xiangming Sun, Chengxin Zhao, et al,. (2020). A new method for directly locating single-event latchups using silicon pixel sensors in a gas detector. Nuclear Inst. and Methods in Physics Research, A, 962, 163697. doi:10.1016/j.nima.2020.163697

[38] R. He, X.Y. Niu, Y. Wang, et al. Advances in nuclear detection and readout techniques[J]. Nuclear Science and Techniques, 2023, 34(12): 205. doi:10.1007/s41365-023-01359-0

[39] Snoeys W. Monolithic CMOS Sensors for high energy physics–challenges and perspectives[J]. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2023: 168678.doi:10.1016/j.nima.2023.168678

[40] Rinella, G. Aglieri, and Alice Collaboration. Developments of stitched monolithic pixel sensors towards the ALICE ITS3. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 1049 (2023): 168018. doi:10.1016/j.nima.2023.168018

[41] Wu T, Li S, Wang W, et al. Beam test of a 180 nm CMOS Pixel Sensor for the CEPC vertex detector[J]. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2024, 1059: 168945.doi:10.1016/j.nima.2023.168945

[42] Liu, Y., Ai, XC., Xiao, GY. et al. Simulation study of BESIII with stitched CMOS pixel detector using ACTS. NUCL SCI TECH 34, 203 (2023). doi:10.1007/s41365-023-01353-6

[43] Gao, C., Huang, G., Sun, X. (2016). Topmetal-II-: a direct charge sensor for high energy physics and imaging applications. Journal of Instrumentation, **11(01)**, C01053–C01053. doi:10.1088/1748-0221/11/01/c01053

[44] W. Ren, W. Zhou, B. You, et al. Topmetal-M: A novel pixel sensor for compact tracking applications[J]. Nuclear Instru-

ments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2020, 981: 164557. doi:10.1016/j.nima.2020.164557

[45] J. Liu, C. Gao, H. Wang, et al. Design and preliminary characterization of a novel silicon charge sensor for the gaseous beam monitor at the CSR external-target experiment[J]. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2023, 1047: 167786. doi:10.1016/j.nima.2022.167786